

APPLICATION OF ARTIFICIAL INTELLIGENCE TO SYSTEMATICS: SYSTEX—A PROTOTYPE EXPERT SYSTEM FOR SPECIES IDENTIFICATION

J. B. WOOLLEY AND N. D. STONE

Texas A&M University, Department of Entomology,
College Station, Texas 77843

Abstract.—The applicability of expert systems (a branch of artificial intelligence) to identification and classification in systematics is examined. Existing devices used for taxonomic identification (dichotomous, tabular, and computerized keys) are reviewed and compared to expert systems. Expert systems technology is briefly reviewed, and a prototype expert system (SYSTEX) for species identification in one species group of the genus *Signiphora* Ashmead (Hymenoptera: Signiphoridae) is presented. SYSTEX is a rule-based, backward chaining system with 114 rules. It was developed using a commercially available expert system shell. In building the identification system and in its use, we find the expert system approach to be in most ways superior to the dichotomous key and other identification devices in terms of efficiency and ease of use, tolerance of missing data, explanatory capability, and the ability to provide meaningful output when an unambiguous identification is not possible. The potential for incorporation of phylogenetic hypotheses into an identification device is discussed. [Diagnosis; identification; classification; expert systems; artificial intelligence; dichotomous key; tabular key; computer identification; *Signiphora*; Signiphoridae.]

The primary tasks of the systematist are the delineation of natural taxa and the determination of their relationships. The starting point for the dissemination of the results of systematic research is often the process of identification, the determination of the identity of an organism to some level in a taxonomic hierarchy. Most systematists perform identifications, and for many this service function is a condition of employment. In many taxa, considerable experience, a large literature library and a comprehensive research collection are required for authoritative identifications. Thus, it is generally the case that as systematists grow in experience and expertise and are better able to attack problems in classification, their skills become increasingly in demand for identification. We perceive a need in the systematics community for more efficient devices for identification. Reducing the time, effort, and expertise required for identification will allow systematists to concentrate more of their efforts on systematic research.

Dichotomous Keys

The traditional device used for the identification of biological specimens is the di-

chotomous key. Most biologists (and all systematists) are well acquainted with the use of dichotomous keys and with their advantages and disadvantages. Guidelines for the preparation of workable dichotomous keys are found in Metcalf (1954), Bohart (1961), and Pankhurst (1978). As these authors point out, the construction of dichotomous keys is among the most difficult tasks in revisionary work. If at a later date, new taxa need to be incorporated into an existing key, fundamental changes in the key's structure are often necessary. In the worst case, the entire key must be rewritten. Several computer programs and utilities are available to aid systematists in the production of dichotomous keys (Morse, 1971, 1974; Pankhurst, 1971; Dallwitz, 1974, 1978; Hall, 1975; Payne, 1975; Ross, 1975).

Dichotomous keys also present many difficulties for the user. Most serious perhaps are the consequences of a wrong choice early in a lengthy key. This can send the inexperienced user on a fruitless search through couplets which never seem really to apply to the specimens at hand. If the user is attempting to identify a taxon which is not included in the key it is possible to obtain an incorrect result with little diffi-

culty, or to encounter couplets for which either choice applies to some degree. The use of ambiguous language or poorly defined character states early in a key can make it virtually unusable until users learn to circumvent the difficult couplets. Most users of keys have had the experience of arriving at a terminal couplet in a long key only to find that neither choice applies. In many cases, this result is useless and one is no closer to an identification than at the outset.

Many of these problems can be avoided in a well designed dichotomous key. Key characters should be chosen so that the interpretation of character states is unambiguous on as many specimens as possible. Although the primary purpose of identification devices is not to represent phylogenetic hypotheses (or other notions of taxonomic structure), we argue that a dichotomous key which embodies hypotheses of taxonomic relationship will be more useful. For example, one might use synapomorphies in the early part of a key to delineate monophyletic subsets of taxa. Then if a user is confident about choices made at initial couplets, the specimens can at least be placed in some natural group of taxa. This can only be done, obviously, if the phylogenetic structure of the group is understood. Another device was used by Noyes and Hayat (1984) in a relatively long key to genera in a family whose phylogenetic structure is not well understood. These authors constructed a key in which groups of not more than 27 couplets always lead to an endpoint. Thus, a user who makes a wrong decision at any point may soon discover the problem.

Dichotomous keys also tend to use monothetic identification criteria. That is, some unique set of character states is both necessary and sufficient to identify an unknown. This does not present a problem if taxa can be recognized by monothetic criteria. For example, taxa which can be recognized by particular synapomorphies will be monothetic, if no characters are homoplastic. However, if taxa are defined by polythetic criteria, no single vector of character states is necessary for identification,

and many character state vectors may apply to a particular taxon. To include variation for character states in a dichotomous key, each taxon with variable character states must be explicitly treated. One commonly used method is to include statements in the key with "or" clauses (state A or state B). However, this requires complete enumeration of all known character states. Alternatively, qualifiers such as "usually," "often," "rarely," etc. are used in the couplets. This inevitably leads to a loss of precision and frustration to the user who has no way of knowing whether or not the "rare" case applies. A third approach to variable character states for a taxon is to identify the taxon at several endpoints in the key. This also requires complete enumeration of all possible combinations of character states, greatly increases the complexity of key construction, and serves to obscure the logic of the key from the user.

Tabular Keys

The tabular key (Newell, 1970, 1972) is an alternative to the dichotomous key. A tabular key is essentially a taxa by character matrix with character states coded for each taxon. If the group is too large to be represented in a single matrix, then the initial matrix provides entry points to subsequent tables in which the subgroups are treated. As well explained by Newell (1970, 1972), this method has many advantages both to the designer of the key and to the user. For example, a certain amount of missing data presents little problem to the user, provided that sufficient data are available to provide a unique combination of character states. A tabular key may also represent the presumed phylogenetic structure of a group, so that an ambiguous or incorrect result may still provide some indication as to the relationships of the material.

Computerized Keys

With the increasing availability of computer resources in the 1960s, systematists began to investigate computerized alternatives to written dichotomous keys. Morse

(1975) and Pankhurst (1978) presented overviews of several approaches to the development of computerized identification devices. In general, emphasis has been placed on access to a formatted data base containing matrices of taxa and associated character state data. One approach (Morse, 1975, and references therein) used a monothetic character set matching method. Essentially, the user inputs a complete vector of character states for an unknown. The program then uses a table look-up procedure to search for an exact match in a character state vector for a known taxon. All logically possible combinations of characters must be included in the table. If several character state vectors apply to a particular taxon, these must all be treated explicitly. The method is polythetic in the sense that no single character state vector is required for identification. As the number of possible character state vectors for "n" binary characters is 2^n , the method rapidly becomes unmanageable for problems requiring many characters. It is apparently very efficient for problems in which a small number of characters are sufficient (Morse, 1975). Matching procedures have also been devised in which a vector of exact character state matches is not required (Pankhurst, 1975). In this case, various measures of similarity (e.g., the simple matching coefficient) were used as the criteria for identification.

Considerable attention has also been given to "polyclave" identification procedures (Morse, 1975), also called multiple entry keys. Here the program has access to a data file containing known taxa and associated character state vectors. The user is free to input observed character states in any order. With each data entry the program searches the data file and attempts to eliminate taxa with other than the observed character states. After each data entry, the program outputs to the user a list of possible taxa remaining. The program can often be queried regarding characters yet to be used and some programs allow backtracking if an error in data entry is suspected (Morse, 1975). In an interactive setting, such a device is a powerful tool for

an experienced user who might recognize rare or unusual character states in an unknown. However, the burden of efficient character choice at each step is on the user.

Wilson and Partridge (1986) provide a recent example of a third type of computer identification program, which they call an "online key." Essentially, an online key is an extension of polyclave methods in which deliberate attempts are made by the program to optimize the sequence of character state input by the user. Characters are weighted in their program using seven formulae, representing criteria for optimal discrimination or ease of use. The program accumulates information on character desirability during each session, and uses this information to select the sequence of queries for user input. The sequence of characters used to arrive at a particular identification may therefore change substantially over time.

The ONLIN3 identification program (Pankhurst and Aitchison, 1975; Pankhurst, 1984) uses a single criterion, Gyllenberg's (1963) separation number, to provide optimized character selection in a polyclave format. With binary characters, this criterion gives higher weight to characters which discriminate between more pairs of taxa. As with other online keys, ONLIN3 provides the ability to check a presumed taxon against the database, using an algorithm to determine an optimized order of character state input. Additional options allow the user to retrieve the information already entered for an unknown and to determine the character states by which an unknown differs from a known taxon.

A program called XPER (Lebbe, 1984; Forget et al., 1986) has recently been developed that is similar to ONLIN3. XPER allows the user to enter more than one character state in response to a query. In a case in which the identity of a taxon is presumed and the program is used to verify this presumption, XPER provides an ordered list of potential discriminating characters, optimized to provide the shortest path to a solution. Ability to edit data files and database management utilities are also

provided. XPER is available to the public in France through the MINITEL system and is now in wide use for mushroom identification (Jaques Lebbe, pers. comm.).

One way of making polyclave methods polythetic is to allow one or more mismatches between character state vectors in the unknown and possible known taxa (Morse, 1975). All of the programs discussed above incorporate this feature. The number of allowable mismatches is set by the user and would appear to be purely arbitrary. To be truly polythetic, an identification device must invoke either representations of taxonomic distance between an unknown and known taxa or use conditional probabilities to represent the likelihood of an unknown being a particular taxon given certain information. The first method was used by Gyllenberg and Niemelä (1975). Essentially, each known taxon occupies a neighborhood in a multidimensional character space, a hyperellipsoid, the boundaries of which are determined by the known ranges of character state variation. An unknown with a particular character state vector represents a point in such a hyperspace. Inclusion in, or phenetic distance to, some known neighborhood are the criteria for identification. Polythetic identification algorithms have also used probabilistic methods based on likelihood estimates of observed character states in an unknown given the frequency of character states in known taxa (Morse, 1975; Willcox and Lepage, 1975).

Polyclave and online methods would appear to have many advantages over dichotomous keys. Polyclave methods would be most useful for the experienced user who can make informed choices for the order of character state entry for efficient discrimination. Online methods shift the burden of character choice to the program by including algorithms for character selection. Both methods embody many of the advantages of a tabular key and have added an efficient look-up device and an interface between the user and the data table. Further descriptive information on particular characters and character states could be

made available as "help" files, thus addressing potential problems of ambiguous characters. If data for certain characters are missing, the user simply uses another character. In such case, the final result might be a group of possible taxa. Whether such taxa would represent a natural group would depend on the nature of the character set and structure of the program.

With the present availability of data base management software, any systematist with a microcomputer has the ability to construct what are essentially monothetic polyclave key devices. By repeated querying of a character state database and the use of Boolean logical operators an unknown could be referred to a single taxon or a set of possible taxa. Here again, the sequence of queries is up to the user. One problem with any such identification device is that it does not degrade gracefully. That is, anomalous or unknown character state combinations may yield an empty set of possible taxa and missing data may produce an artificial assemblage of possible taxa.

Status of Identification Devices

We can identify three trends in the development of alternatives to written dichotomous keys. One trend is an increasing reliance on computer programs for quick comparison of observed character state vectors with those known for a given set of taxa. Either through data base systems or applications of statistical-decision theory, computers now provide useful tools for identification. The second trend, exemplified by Newell's (1970) tabular keys and to some extent by polyclave and online keys, is making the logical structure of identification devices increasingly flexible and transparent to the user. Such devices have advantages when the user has missing, imprecise, or variable character state data for an unknown. The third trend is an increased focus on providing users with the shortest or most efficient path to identification.

The merging of these trends is now possible through building expert systems for taxonomic diagnosis (Stone et al., 1986).

Diagnostic expert systems are computer programs which use judgment and logic derived from human experts to address problems in identification based on facts as well as uncertain information. Expert systems are particularly well suited to capturing relevant knowledge about a taxonomic group (from a recognized expert), and making that expertise readily available to systematists worldwide. Exactly what makes expert systems special and why they may be preferable to dichotomous keys and other devices as identification tools is discussed below.

EXPERT SYSTEMS

The field of expert systems, like the field of robotics, is derived from artificial intelligence research and thus incorporates symbolic and nonalgorithmic methods of problem solving (Buchanan and Shortliffe, 1984). Expert systems solve real world problems—problems that require judgment, experience, and scarce expertise for their solutions. They do so by encoding the knowledge and experience of a human expert into computer code (a *knowledge base*), which is accessed during a consultation by a computer program capable of making inferences based on that knowledge. In effect, the user of such a system addresses the computer in much the same way she would address an expert.

Expert systems, when well designed, can be tremendously useful. The DENDRAL system developed at Stanford University in the late 1960s and early 1970s (Feigenbaum et al., 1971) to generate spatial representations of organic molecules based on various data (e.g., mass spectrogram and NMR) was so successful that its results have been published in several papers and it has acquired the reputation of being faster and more accurate than many experts in the field (Duda and Gaschnig, 1981). The XCON system developed by Digital Equipment Corporation to configure VAX 11/series computers to customer specifications has also proven to be more reliable than were the human experts themselves (Kraft, 1984).

However, not all problems are appro-

priate domains for expert systems. Expert systems do not provide new information. They are best used as integrators of knowledge and as delivery devices for scarce expertise and training. Duda and Gaschnig (1981) list four prerequisites for developing a successful knowledge-based expert system:

- 1) there must be at least one person, generally acknowledged to be an expert, who can perform the assigned task well;
- 2) the expert's ability to perform the task must be primarily due to special knowledge, judgment, and experience;
- 3) the expert must be capable of explaining the thought process he uses in applying his special knowledge to a problem; and,
- 4) the task must be bounded in scope, not open-ended.

The general problem of taxonomic identification fits well the prerequisites of Duda and Gaschnig. Systematists are generally acknowledged as experts who have acquired their knowledge through years of singular experience (1 and 2 above). The problem domain for taxonomic identification is naturally bounded by the number of known species and taxonomic structure in a group (4 above). Furthermore, most systematists can make explicit the logical basis for their method of classification and identification. Thus prerequisite number three is generally satisfied.

Structure of an Expert System

Before describing our prototype expert system it is necessary to explain some of the basics about expert system structure. The treatment here will be somewhat brief, hence the reader is encouraged to look elsewhere for a more thorough review of expert systems (see the April 1985 edition of BYTE magazine for an excellent series of articles).

Virtually all expert systems developed for diagnosis to date are classified as production systems (Davis and King, 1984). They all contain three components: a knowledge base, an inference engine, and a working global memory (Fig. 1). The

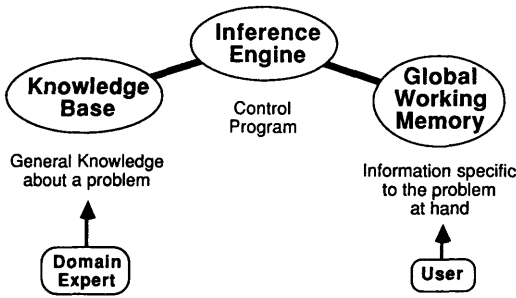


FIG. 1. The three components of a rule-based expert system or production system. The knowledge base is a database of logically independent rules which summarize the knowledge of an expert about a particular problem domain. The user supplies specific information to the working memory about her problem while interacting with the system. The inference engine controls the selection, testing, and execution of rules that relate to the user's problem.

knowledge base is generally composed of rules and facts which together represent the experience and reasoning logic of the domain expert(s). Rules are *antecedent-consequent* clauses (Duda and Gaschnig, 1981) or *if-then* statements; they have a left-hand side and a right-hand side. The left contains conditional statements (antecedents) that together define a pattern or hypothetical "state of the world"; the right contains instructions to be carried out (consequents) in the event that the current state of the world matches the hypothetical pattern described in the left-hand side (Fig. 2).

The current state of the world is contained in the working global memory. Generally, the information reflects knowledge specific to the question at hand as supplied by the user. However, the information in memory can be supplemented or modified by the consequents of rules that are found to apply to the current pattern. Thus the state of the world is dynamic; the action of one rule will immediately affect the applicability of other rules by altering the information stored in memory.

The process of searching through the knowledge base for rules that apply to the current state of the world, including the particular strategy used in searching and

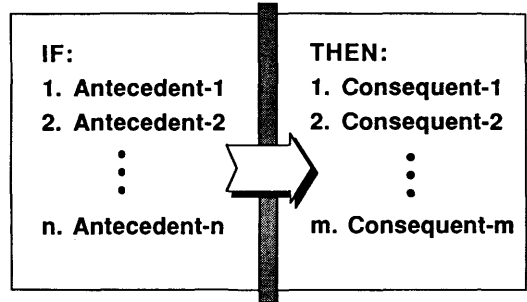


FIG. 2. A generalized rule showing the left-hand and right-hand sides. When the antecedents are satisfied, the consequents become true (i.e., they are executed). Note that a single rule may have multiple (n) antecedents and/or multiple (m) consequents.

applying rules, is the task of the inference engine. In effect, the inference engine is the control program, manipulating the knowledge base, updating the state of the world and remembering the chain of reasoning being used. There are two principal control strategies used in expert systems (see Fig. 3): forward-chaining (data-driven rule selection) and backward-chaining (goal-driven rule selection). Both of these strategies employ a form of pattern matching. That is, information in global memory represents a pattern of knowledge that must match the pattern specified in each rule's left-hand side before that rule can be executed.

A forward-chaining inference engine

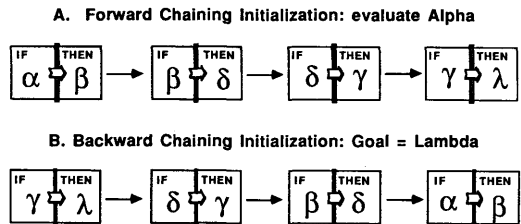


FIG. 3. Two common inference control strategies operating on the same four-rule knowledge base. Forward-chaining in A is data driven; it lets the information known to the system dictate which rules are appropriate. Backward-chaining in B is goal driven; it picks rules according to the goal it is attempting to achieve. Here, each strategy requires one question about alpha to make a conclusion about lambda. See discussion in text for further explanation.

proceeds by searching the rules for one whose left-hand side describes a pattern matching the current state of the world. Such a rule is then applied (*fired*) and the global memory is updated according to the rule's consequents. The updated global memory represents a new state of the world, and the process is repeated. As shown in Figure 3A, forward-chaining inference engines have a predefined starting point to initialize the system with some data about the current problem. In the example in Figure 3, the inference engine begins by asking the user the status of alpha. Once it is known that alpha is true, the logical steps (shown as elementary rules) quickly lead the system to conclude that lambda is true as well. The main selection criterion for rules is thus the data stored in global memory, hence the term, *data-driven*.

Backward-chaining works in the opposite direction: instead of letting the incoming data dictate the inference procedure, the inference engine selects a goal at the outset and seeks rules whose consequents result in that goal being achieved. When such a rule is found, the conditions set forth in the antecedents are tested against what is known in the global memory. If the conditions are all met, the rule succeeds (i.e., the goal is achieved). Otherwise, if some set of antecedents must be evaluated before the system can determine whether all the conditions are met, then the primary goal of the inference engine is set aside temporarily (but saved in a memory stack), and a new goal selected by the inference engine, namely to evaluate the unknown antecedents. This process continues recursively until the problem is solved (the primary goal is achieved). In the example shown in Figure 3B, the goal of the inference engine is to show that lambda is true. The inference engine finds that to prove that lambda is true it must know that gamma is true. Showing that gamma is true becomes the new goal, and the inference engine finds quickly that to prove gamma is true it must know that delta is true, etc. After two more steps, the inference engine needs to know the value of alpha. At this

point it asks the user the state of alpha, and if the user tells the system that alpha is true, the inference engine knows that lambda is true. Because rule selection in this strategy depends mainly on the goals selected by the inference engine, it is termed *goal-driven*.

Notice from the example above (Fig. 3) that to the user there is no perceptible difference between systems using forward- and backward-chaining. Both systems asked one question regarding the state of alpha, and both concluded that lambda was true when alpha was true. Thus despite the fact that an expert system uses a backward-chaining control strategy, one can freely interpret the rules in the knowledge base as if the system employed forward-chaining (i.e., in an *if-to-then* order).

Features of Diagnostic Expert Systems

The production system design and unstructured flow of control of rule-based expert systems endows them with several powerful abilities and features not found in conventional computer applications. Among these are the following.

1) *The ability to mimic human reasoning.*—The central feature of expert systems is that they proceed toward a solution in a way patterned after a human being. The result is that the logical progress of the system is generally easily understood by its users. This is primarily due to the "pattern matching" flow of control used by the inference engine. Expert systems, like human beings, make inferences based on recognizing a familiar pattern within the overall problem domain.

2) *Ability to incorporate uncertainty.*—Very seldom do human experts rely solely on conclusive evidence in solving a complex problem. Judgment and experience play a key role in weighing evidence that is imprecise or vague. When experts are asked to describe the rules they use in making decisions, they often must speak in terms of probabilities or confidence in their inferences and deductions. Expert systems incorporate this imprecision by associating probabilities or certainty factors with the antecedents and consequents of rules. This

allows the reasoning process and ultimate advice of the system to be expressed in terms of probabilities. For example, the expert system PROSPECTOR, designed to locate potential mining sites for various types of minerals (Duda et al., 1979), uses two measures of uncertainty: one to rate the confidence that a piece of evidence supports a given conclusion, a second to rate how detrimental the lack of that evidence would be to the conclusion. Thus, internal confidence factors can be used to examine both α and β errors.

3) *Explanation facility*.—Because rules represent discrete logical steps in the deduction process of an expert, and because the inference process of an expert system produces, in essence, a sequential list of rules fired, the history of a session with an expert system can be thought of as a sequence of logical deductions. Simply by reciting that history at the request of the user, an expert system can explain its reasoning in easy to follow logical steps. This ability is present in virtually all expert systems and is one of the clearest advantages of building expert systems.

4) *Ability to build generalizable systems*.—The particular problems experts solve are far more numerous than the types of problems they solve. One would expect that the problem of identification of species of *Signiphora* Ashmead (Hymenoptera: Signiphoridae), for example, is at least conceptually little different from the identification of species in unrelated genera. The particular knowledge necessary to carry out reliable identifications, however, will certainly be different. Expert systems are naturally structured to be generalizable to problems of a similar type. The inference engine, and the form of the rules in the knowledge base can easily be transferred to related problem domains. It is only the specific antecedents and consequents in the rules that are unique to the problem domain of the original system.

SYSTEX

We designed SYSTEX (SYSTematics EXPert) as a pilot project to test the application of expert system technology to the

general problem of taxonomic diagnosis. In choosing a model system, we looked for a relatively small group which was well worked out taxonomically. Identification in the group should be possible by an experienced expert. We also wanted to work with a group of species that would present difficulties in identification for a skilled specialist who did not happen to have experience with these particular taxa.

The *aleyrodids* species group in the genus *Signiphora* possesses these attributes. *Signiphora* is being revised by one of us (Woolley, unpubl.). The *aleyrodids* group contains 12 species in the New World, 3 of which are undescribed. Some species are very similar morphologically but generally can be distinguished from one another by unique combinations of diagnostic characters. Considerable effort had been devoted to the choice of diagnostic characters during the revisionary work, thus we had a workable starting point. Even so, some characters are difficult to observe or interpret on some specimens and some pairs of species are difficult to separate due to overlapping distributions of character states. Phylogenetic structure within the *aleyrodids* species group is not well understood and the diagnostic criteria for most species are therefore polythetic. In some cases, host records or other biological characteristics, and geographical distributions provide useful criteria for identification.

Diagnostic characters in the *aleyrodids* species group involve patterns of coloration on the fore wings, antennae, and body; presence, absence, or relative length of setae on the fore wing marginal vein, number of setae on thoracic sclerites, shape and relative size of abdominal terga and antennal segments, and type of sculpture pattern on the head or thorax. Characters were either binary, meristic, or character states were coded into classes useful in discriminating between taxa. Characters for which the distribution of known states is continuous, for example, the ratio of the lengths of the second and third abdominal terga, were also coded as multi-state characters with discrete classes. A total of 22 characters was used. A virtually complete

taxa by characters matrix for all characters across all species was available at the outset of the project. In addition, we had access to a great deal of information on hosts, species distributions, and data on the biology of particular species.

SYSTEX was developed and delivered using the M1 expert system development software package (Teknowledge Corporation) running on an IBM PC-XT microcomputer. M1 is a backward-chaining, rule based expert system shell with some capability to forward chain. In its documentation for M1, Teknowledge recommends that appropriate uses for M1 would include problems that can normally be solved in a question and answer framework in which an expert interviews the person with the problem. In practice, we found that memory and speed constraints limited the size of the system to about 200 knowledge base entries. The program does provide a primitive means to overlay rule base modules, but the sacrifice in execution time was prohibitive for our application.

We chose a backward-chaining expert system shell both because the M1 package was available, and because our application was well suited to goal-driven logic. Forward-chaining logic is well suited to problems with specific entry points and a wide variety of goals either in terms of numbers or types. However, since taxonomic identification proceeds from the open-ended problem, "where to begin?" to relatively few discrete goals all of the same type (taxa), we chose to begin with the better defined part of the problem, the goals, and to proceed using a backward-chaining strategy.

Rules were derived from a series of consultations between the domain expert (J.B.W.) and knowledge engineer (N.D.S.). The object of these sessions was to develop a rule-based representation of the reasoning process of the domain expert during identification. In practice, this was an iterative process involving construction of preliminary expert systems followed by verification of their accuracy and subsequent modifications to the rule base. During this process, the complete matrix of taxa by diagnostic characters was useful.

SYSTEX contains in its knowledge base a few facts which delineate the scope of the system, and 114 rules that together make up the system's reasoning process. That is, there are flow of control rules, advice rules, discrimination rules, and verification rules. These are not explicitly segregated in the program; however, they were written so that the program would behave in a somewhat modular way. The overall logic of the system is defined by its advice rules and is summarized by the dependency network shown in Figure 4. Notice that the rules concerned with discrimination of species based on particular character states are considered as a module in the decision point called, "What species remain after elimination?" The system invokes those rules only when necessary by directly manipulating the goal that the M1 inference engine is seeking.

Sessions with SYSTEX

There is a fundamental difference between an expert system for species identification and all other standard methods of species identification that depend on keys or database operations. The goal of an identification key is to lead users through a process that generally allows the user to identify any given specimen. The goal of an expert system is to capture the systematist's knowledge, experience, and reasoning process during an identification in a computer program. The resulting program, therefore, has many of the advantages that a systematist herself would have over a key in trying to identify a specimen.

Each session with SYSTEX reflects the logical paths in the dependency network shown in Figure 4. The actual flow of the program is summarized in Figure 5 but in the absence of conflicts can be summarized by the sequence: *assumption—elimination—verification—advice*. The program begins by asking questions about the host from which the specimen was reared and gross body coloration—characters that would be immediately obvious to anyone upon presentation of a labeled specimen. From this preliminary evidence the system makes an initial guess about which are likely species

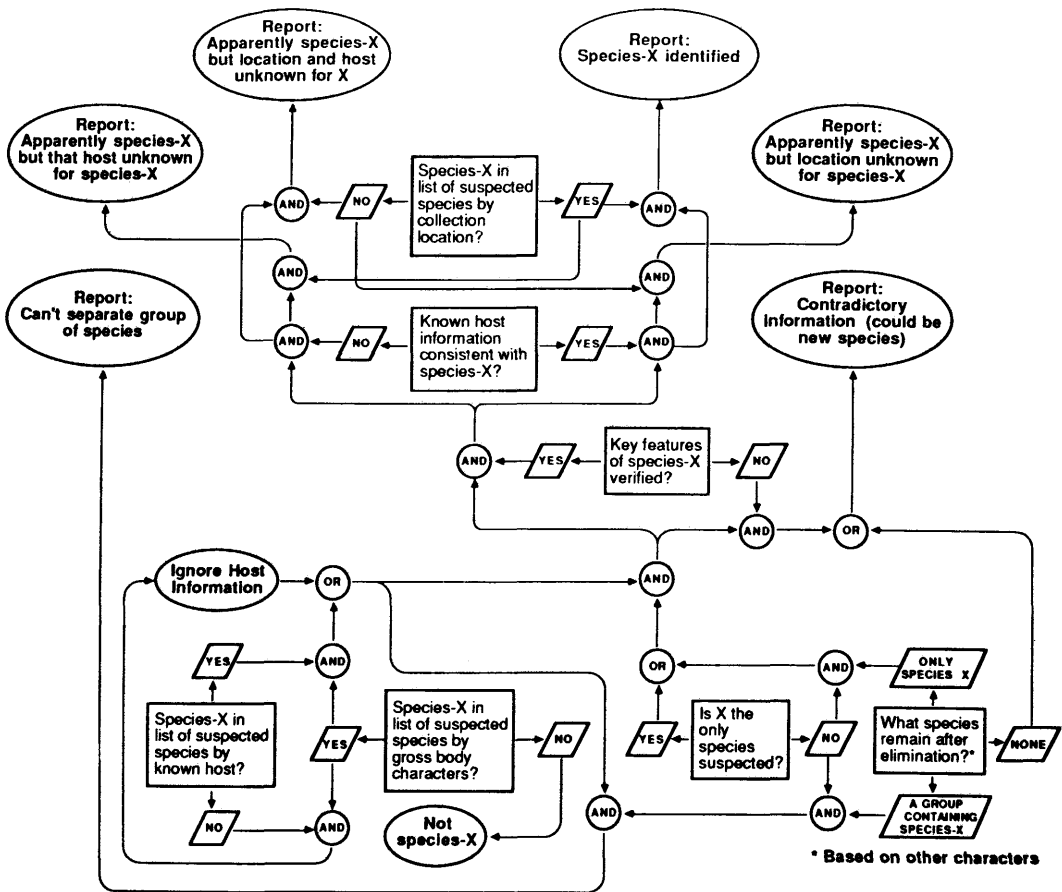


FIG. 4. A dependency network for SYSTEX showing the process followed for each species tested. Arrows are logical paths leading to advice (Reports). Rectangles are variables, parallelograms are possible values of the variables, circles are logical connectors between values of variables, and ovals are conclusions or actions taken by the system. For any given report to be given, the logical path leading to it must be evaluated as true, including all paths leading into each AND node and any single path leading to an OR node. Paths are true when the values of variables in rectangles at the beginning of each path have the value specified in the parallelogram immediately following. (Modified from Stone et al., 1986.)

and then attempts to prove that the specimen is, in fact, one of those likely species.

Narrowing down the list of likely species involves a set of rules called elimination and discrimination rules (see Fig. 6). These rules, like the verification rules discussed below, involve character state values in their antecedents, and therefore, it is when the inference engine is considering these rules that most of the questions to the user are asked. Elimination or discrimination rules are executed only if their antecedents are satisfied. Therefore, the flow of pro-

gram execution through the rule base is dependent upon the particular information which has been input to the program. A variety of pathways through the rule base can lead to any particular end result. This is seen by the user as the program asking different questions or questions in a different order depending upon the nature of the data which had been input. Assuming that the system is successful in reducing its list of likely species to one, the system then declares its preliminary identification and invokes the verification rules.

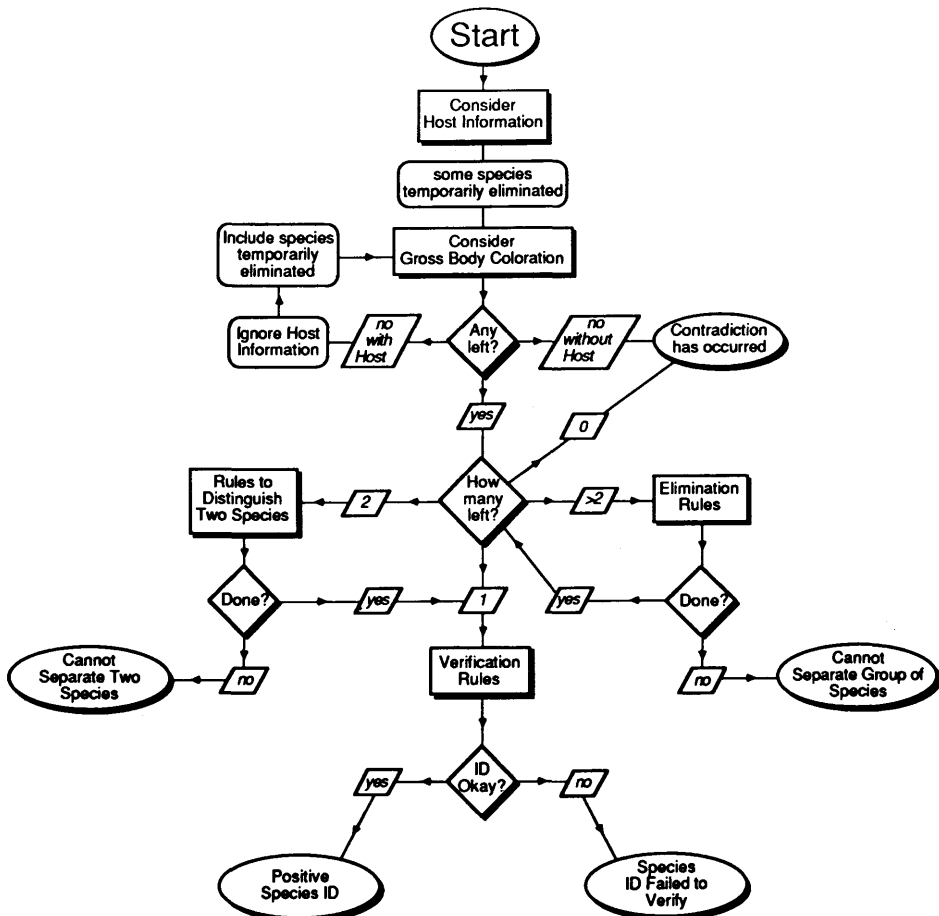


FIG. 5. A flow chart of SYSTEX describing the sequence in which rule modules are invoked by the inference engine. The general flow of control is determined in part by a set of flow-of-control rules in the knowledge base. A more specific flow diagram is not possible to draw because of the nature of expert systems. There is, in fact, no real flow of control in rule-based expert systems. This diagram is thus descriptive, not prescriptive, of how the system behaves.

Verification involves the system's determination that all the diagnostic characters for the assumed species have been verified. Naturally, some of the diagnostic characters will already be known to the system at this stage from processing the elimination rules. However, the system will often have made a preliminary identification based on just a few characters. In this latter case, the system will ask the user to check for the as yet unknown diagnostic characters before making its recommendation definite. Confidence factors can play an important role here in determining how

many and which diagnostic characters permit the system to make a positive identification.

The last step in the verification process is the comparison of the collection location and host information with that known for the identified species. When the location or host are anomalous, the species identification is given with a warning that the host and/or locality of the specimen has not previously been observed. In such cases, the final advice of the system is to contact the expert directly with such important information. Other advice the system can give

Rule m1g-12:

if initial_group = m1_group and (state(<i>aleyrodus</i>) = yes or state(<i>townsendi</i>) = yes) and (state(<i>coquilletti</i>) = yes or state(<i>flavella</i>) = yes) and club-scapel = high and do(reset state(<i>aleyrodus</i>)) and do(reset state(<i>townsendi</i>)) and do(set state(<i>aleyrodus</i>) = no) and do(set state(<i>townsendi</i>) = no) then eliminated = <i>aleyrodus</i> and eliminated = <i>townsendi</i>	; No distinctive coloration and seta M1 present ; state(species) is yes if it is still under ; consideration (has not been eliminated) ; The diagnostic character. ; a high ratio is never found in <i>S. aleyrodus</i> ; or <i>S. townsendi</i> , so they can be eliminated ; if this condition is found. ; elimination consists of resetting the state() ; variable to 'no' and adding the species ; names to the list, 'eliminated'.
---	--

FIG. 6. An elimination rule from the code of SYSTEX. This rule can eliminate *S. aleyrodus* and/or *S. townsendi* Ashmead when either (or both) of these species is still being considered along with one or both of *S. coquilletti* and *S. flavella* Girault.

is shown in Figure 4. Rules leading to advice resulting from conflicting information or failure of the system to narrow down the probable species to a single species name make up the rest of the program.

The example in Figure 7 illustrates the logical flow and ultimate advice of the system as just described and points out some of the strengths of the expert system format. The first three questions concern the host and general body coloration of the specimen. In this case, the host information is missing, but the body coloration is distinctive enough to allow the system to narrow down its initial guess to two predominantly dark colored species, *S. merceti* Malenotti and *S. maculata* Girault (it has already been determined that the species is in the *aleyrodus* species group). At this point, the system asks about characters that are useful in distinguishing between these two species.

This strategy is consistent with the way that experts typically do identifications. Experts tend to make preliminary judgments about the most likely identification for a series or specimen based solely on their host, collection location, and general appearance. The expert then checks to see if her first impression is correct. The same logical process occurs in SYSTEX, allowing it seemingly to leap to conclusions at times, as in this example. SYSTEX makes educat-

ed guesses and then proceeds to verify them, if possible, rather than discriminating the user's species from all known taxa based on a prescribed sequence of character state tests.

The user, when the system asks a specific question relating to the size of a particular seta on the marginal vein of the specimen, justifiably wants to know why the system has jumped from very general to very specific questions. Instead of answering, she asks, "why." After the system explains its reasoning, the user admits that she is unsure of this character, perhaps because the wings are damaged or missing. She answers, "unknown."

The system, because it is pursuing a logical strategy (to distinguish between two likely species), does not shift gears just because the user cannot provide information. The inference engine searches for other rules which will distinguish these species. It is successful in its second choice of characters, and the identification process proceeds to the verification step after the system displays its tentative conclusion, "Almost certain it's *maculata*." Again, this logical process mimics what an expert would do. If an expert believes that the specimen under her lens is species A, but if a character is obscured that she would ordinarily check, she would still pursue the hypothesis that it is species A by taking


```

>> go.
From what type of host was this specimen or series collected?
Answer (diapsid, coolid, or whitely):
>> unknown.

Describe the overall body color (thorax and gaster) as:
"uniform" — All one color, either dark or pale
"mixed" — A mixture of dark and pale or yellow
Answer (uniform or mixed):
>> uniform.

Is the color of the thorax:
"dark" — Uniformly brown or black
"pale" — Uniformly pale or yellow
Choose either dark or pale:
>> dark.

Which best describes the relative length of seta M3?
"short" — Slightly longer than width of marginal vein
"medium" — about 1/3 the length of the marginal vein
"long" — 1/2 to 3/4 the length of the marginal vein
"very-long" — subequal to or longer than the marginal vein
Choose from: short, medium, long, and very-long:
>> why.

Current goal:
distinguish merceti from maculata
Reasoning:
because the body coloration is uniformly dark
only two species are likely (merceti and maculata).
Rule:
if seta-M3 is short
the species is merceti.

Which best describes the relative length of seta M3?
"short" — Slightly longer than width of marginal vein
"medium" — about 1/3 the length of the marginal vein
"long" — 1/2 to 3/4 the length of the marginal vein
"very-long" — subequal to or longer than the marginal vein
Choose from: short, medium, long, and very-long:
>> unknown.

Is the coloration of the fore wing:
"normal" — Infuscated to below the distal end of the
          stigmal vein with hyaline areas present at
          the wing base
"darkened" — infuscated from base to apex
Answer (normal or darkened):
>> normal.

Almost certain it's maculata.
Verifying diagnostic characters...

Are the mandibular ducts parallel-sided or enlarged apically?
Answer (parallel or enlarged):
>> parallel.

From which general area below did the specimen(s) come?
"southwest" — California, Baja Calif., Arizona, New Mexico,
              and Texas
"east" — the eastern United States
"mexico" — Mexico, excluding Baja Calif. and the
          neotropical regions
"neotropics" — the Neotropics, including the West Indies and
              Mexico south and east of the Isthmus of
              Tehuantepec

Choose southwest, east, mexico, or neotropics:
>> neotropics.

Species is Signiphora maculata Girault.

```

FIG. 7. A session with SYSTEX in which there is no host information. Note in this case that the identification proceeds very quickly from the initial questions about body color to the final advice. User's responses are in boldface.

another tack. However, a person using a dichotomous key can be left adrift if a character called for deep in the key were missing or damaged.

The example session ends with the system successfully verifying its initial guess, and the identification being displayed. A

second example (Fig. 8) shows a case in which the system cannot make a certain identification but is still able to narrow the possibilities to two and provide confidence values for the two choices. This example demonstrates how expert systems can use probabilities of character state distributions in a meaningful way to include characters which are polymorphic in taxa.

For example, say that 70% of the specimens of taxon A have been observed to have state 1 for a particular character and 30% have been observed to have state 2. For taxon B, 90% of observed specimens display state 2 and 10% have displayed state 1. If the user has a specimen with state 1, based on this character alone there is a high probability (but less than unity) that the specimen is referable to taxon A. This character can be included in a rule base with confidence factors associated with character states which represent the observed probabilities of character state distributions in known taxa.

This is, in fact, the method used in writing some of the rules to distinguish *S. alcyrodis* Ashmead from *S. coquillettii* Ashmead. In the example shown (Fig. 8), the system identifies the specimen(s) as *S. coquillettii* with a confidence factor of 82%, which represents the observed distributions of the character states involved.

THE FUTURE OF EXPERT SYSTEMS IN SYSTEMATICS

In the future, expert systems may become widely used by systematists with major identification responsibilities. We have already seen that in cases of species identification based on polythetic criteria, expert systems show considerable promise. However, expert systems can also be used to address more complex issues in identification, such as recognizing potential new species or taxa and suggesting how they might fit into a classification or recognizing problems with existing criteria for identification. These types of problems are well suited to expert systems that go beyond a simple rule-based structure to incorporate knowledge about the intrinsic

>> go.

From what type of host was this specimen or series collected?

Answer (diaspidid, coccid, or whitefly):

>> whitefly.

Describe the overall body color (thorax and gaster) as:

"uniform" — All one color, either dark or pale

"mixed" — A mixture of dark and pale or yellow

Answer (uniform or mixed):

>> mixed.

Is the color of the thorax:

"dark" — Uniformly brown or black

"mixed" — A mixture of dark and pale (yellow)

"pale" — Uniformly pale or yellow

Choose either dark or pale:

>> mixed.

Is the coloration of the propodeum:

"dark" — Uniformly brown or black

"mixed" — A mixture of dark and pale

"light" — Uniformly pale or yellow

Answer (dark, mixed, light):

>> light.

Is seta M1 on the fore wing marginal vein present (yes or no)?

>> yes.

Is the sculpturing on the mesoscutum and vertex transversely imbricate or reticulate?

Answer (imbricate or reticulate):

>> Imbricate.

Describe the coloration of the antennal club:

"uniform" — Uniformly pale or brown to apex

"half" — Dusky in the distal half

"third" — Dusky in the distal 1/3 or 1/4

"sixth" — Dusky in the distal 1/6

Pick uniform, half, third, or sixth:

>> uniform.

Measure the lengths of the antennal club and scape. Is the ratio of club length to scape length:

"high" — (ratio > 1.75)

"medium" — (1.5 <= ratio <= 1.75)

"low" — (ratio < 1.5)

Answer (high, medium, or low):

>> why.

Current goal:

eliminate *coquillettii*

Reasoning:

because the body coloration is not distinctive, and seta-M1 is present, only the following species are likely:

aleyrodia, *coquillettii*, *townsendi*, *flavella*.

coquillettii has not been eliminated and at least one of the other three species has not been eliminated.

Rule:

if the ratio of the antennal club to scape length is low (i.e., less than 1.5), then we can eliminate *coquillettii*.

Measure the lengths of the antennal club and scape. Is the ratio of club length to scape length:

"high" — (ratio > 1.75)

"medium" — (1.5 <= ratio <= 1.75)

"low" — (ratio < 1.5)

Answer (high, medium, or low):

>> medium.

Determine the ratio of the length of the second abdominal tergum to the length of the third abdominal tergum.

Is it:

"high" — (ratio >= 0.6)

"medium" — (0.4 < ratio < 0.6)

"low" — (ratio <= 0.4)

Answer (high, medium, or low):

>> why.

Current Goal:

distinguish *aleyrodia* from *coquillettii*

Reasoning:

since the body coloration is not distinctive, and seta-M1 is present, and *townsendi* and *flavella* have been eliminated, only two species are likely (*aleyrodia* and *coquillettii*).

Rule:

if the ratio of antennal club length to scape length is medium and the ratio of the lengths of TII and TIII is low, then the species is going to be *coquillettii* 82% of the time based on the known distribution of these characters.

Determine the ratio of the length of the second abdominal tergum to the length of the third abdominal tergum.

Is it:

"high" — (ratio >= 0.6)

"medium" — (0.4 < ratio < 0.6)

"low" — (ratio <= 0.4)

Answer (high, medium, or low):

>> low.

I cannot separate the following species.

probable id = *Signiphora coquillettii* Ashmead (82%).

probable id = *Signiphora aleyrodia* Ashmead (18%).

FIG. 8. A second session with SYSTEX. More information is known here than in the previous example (Fig. 7), but the system is unable to identify the species with 100% certainty. User's responses are in boldface.

interrelationships of the variables in the knowledge base.

Deep Versus Surface Representation of Knowledge

The way knowledge is represented in expert systems is an area of intense interest to artificial intelligence researchers. For this discussion, it is sufficient to consider the difference between surface representation and deep representation.

Surface representation defines only the consequences of knowing the values of

items in the knowledge base—no deeper interrelationships between items are included. Most expert systems (including SYSTEX) use surface representation of knowledge in the form of rules as described above. In the case of organisms, the starting point for developing a knowledge base using surface representation will be, in most cases, a taxa by characters matrix which will uniquely define taxa in a particular group. Such a data structure is very similar to what is called a "decision table," which can be converted to a rule base either

by a knowledge engineer or by automatic rule induction algorithms (Quinlan, 1982; Pankhurst, 1983). A rule base derived from such data will represent relationships among taxa only to the extent that the character states themselves indicate relationship. For example, if particular character states are synapomorphies which allow one to recognize membership in monophyletic groups, the rule base could be constructed to assign unknowns to natural taxa.

Deep representation is used for knowledge domains in which there is an underlying structure to the data which can be explicitly defined and modeled. When items in the knowledge base are linked by deep representation, they are organized into sets of related items, with the relationships specified. When the relationships are stored in a data structure along with the items, the result is called a frame; when the items are arranged as the nodes of a branching network with the interconnections defining the relationships, the result is called a semantic net. In either case, the deep structure enables the expert system to make analogies and abstractions by understanding how different types of information are related.

Incorporating Phylogenies in the Knowledge Base

An expert system using surface representation might attempt a diagnosis by looking for unique combinations of diagnostic characters. However, if the specimen represents an unknown taxon or a known taxon with an unknown vector of character states, the diagnosis might fail. Alternatively, if the user is missing data, the information that can be provided might indicate a set of possible taxa. Whether or not this set of possible taxa represents a natural group of taxa (i.e., a monophyletic group) depends on the nature of the character set used for surface representation.

Deep representation offers a means to make diagnostic expert systems more robust to missing or incorrect data. The semantic net, in particular, is used for knowledge domains in which a hierarchical tree structure can represent patterns in the data.

Semantic nets have a direct application in systematics, since phylogenetic hypotheses (cladograms or sequenced classifications) and supporting evidence (synapomorphies) can be explicitly represented in the knowledge base. An expert system attempting an identification using a semantic net might begin by establishing to what lineages and sublineages the material is referable by testing for the presence or absence of particular apomorphic character states. If the group is completely resolved phylogenetically (i.e., all taxa are terminal nodes in a completely resolved cladogram) such a process could continue until a unique determination resulted. In a case in which an unambiguous identification is not possible with available data, the expert system could still assign the unknown specimens to some lineage.

Deep knowledge could be used as a backup to surface knowledge. For example, the program might attempt a rapid identification based on unique or diagnostic character states. If that failed due to missing data, the program might then establish to what lineage the taxon is referable. Deep representation will also be useful in cases in which conflicting or unknown combinations of characters have been input to the program. Because phylogenetic hypotheses are nested sets of character state distributions, particular character state vectors which are not consistent with known patterns can be quickly singled out for reanalysis by the user. The program might establish to what lineage the unknown is referable based on the bulk of the evidence by finding the largest subset of observed character states consistent with established patterns. A compatibility or parsimony algorithm could be invoked to accomplish this task with some degree of certainty if necessary. Conflicting character states would be pointed out for re-examination by the user. It may be that the user has scored and input the character states correctly and that the phylogenetic hypothesis itself must be re-examined in the face of new data. In such a case, the expert system could report this result to the domain expert(s) or log each session

TABLE 1. Characteristics of various devices for identification. For a more complete discussion of each attribute and type of device, refer to the text.

Attribute	Dichotomous keys	Tabular keys	Computerized keys				
			Matching methods	Polyclave methods	Polythetic methods ^a	Online methods	Expert systems
Structural efficiency	Design ^b	User ^c	No	User	Some ^d	System ^e	System
Dynamic efficiency	No	User	No	User	No	Some	System
Missing data	Design	Yes	Yes	Yes	Yes	Yes	Yes
Graceful degradation	No	User	Yes	Yes	Yes	Yes	Yes
Explicit probabilities	No	No	No	No	Yes	No	Yes
Easily updated	No	Yes	Yes	Yes	No	Yes	No
Transparent logic	Design	User	No	User	No	No	Yes

^a Includes probabilistic and phenetic distance methods.

^b Can be designed into the key.

^c Must be supplied by user.

^d Some applications have this attribute.

^e Supplied by the system or program.

with such anomalies to a data file. Obviously, such feedback would be of tremendous value in testing and improving classifications.

DISCUSSION

It will be useful at this point to summarize the attributes we believe an optimal identification device should possess. First, each query for data from the user should be unambiguous, both in the wording of the query and in the interpretation of character states on the specimens. Second, the likelihood of obtaining an incorrect result with correct responses to queries should be reduced to a minimum. Any identification device should have both of these attributes, and we will not discuss them further. A number of other desirable features of identification devices are possessed by some devices only:

1) *Structural efficiency*.—The device should be designed so that only information directly pertinent to a particular identification is required.

2) *Dynamic efficiency*.—The device should also be dynamically efficient; i.e., the device or the user should be able periodically to determine which additional characters will provide the most efficient progress towards an identification.

3) *Missing data tolerated*.—The device should be tolerant of a certain amount of missing data.

4) *Graceful degradation*.—If more than one taxon is possible given the input data, the

device should provide the set of possible outcomes. If no taxon is possible given the input data, the device should provide the taxon or taxa to which the unknown is closest, and indicate by what characters the unknown differs.

5) *Explicit probabilities*.—If more than one taxon is possible, the device should provide the probabilities, based on explicit criteria, associated with each possible outcome.

6) *Easily updated*.—One should be able to incorporate new taxa without a major restructuring of the identification device.

7) *Transparent logic*.—At any point in an identification session, the user should be able to retrace the logical pathway already followed and assess the current status of the identification.

The extent to which various devices for identification possess these attributes is shown in Table 1. Expert systems can be designed to seek a rapid or efficient pathway for identification, and this pathway is dependent upon the information available to the system at any given time. An identification may proceed by an efficient pathway even in the hands of an inexperienced user or one unfamiliar with taxonomic structure in the group. SYSTEX uses a heuristic search strategy, in which initial data input is used to choose a small set of taxa for evaluation. If the system's initial "guess" proves to be incorrect, a wider set of possibilities is evaluated. Dichotomous keys can be designed to be structurally ef-

ficient, but since they involve a fixed set of pathways to each terminal couplet, they cannot be dynamically efficient. Other devices, such as tabular and polyclave keys, can be dynamically efficient in the hands of an experienced user. Online keys contain algorithms specifically designed to provide dynamic efficiency.

All devices except dichotomous keys tolerate some missing data. Missing data can also be tolerated in dichotomous keys if more than one character is included in each couplet. Ambiguous or incorrect data would seem to pose a greater problem in devices with an elimination strategy, such as dichotomous and polyclave keys, than in devices which tend to make a large number of comparisons simultaneously, such as matching or polythetic devices. Expert systems can be designed to accept ambiguous data in the form of multiple character states (some computer keys also accept multiple entries), or character states with associated confidence factors (e.g., 50% confidence that character 1 has state *a*). The verification procedure in SYSTEX is a means to overcome the effects of ambiguous or incorrect data entry by requiring agreement in multiple diagnostic characters for an identification.

Expert systems degrade gracefully. If the input data is not sufficient for a unique determination, an expert system can produce the set of taxa to which an unknown may be referable. In SYSTEX, if the combination of character states, host, and geographic locality does not correspond to any known taxon, the user is alerted to the problem. Most other computerized devices which use matching or distance criteria for identification have a similar capability. If the knowledge base of an expert system incorporates phylogenetic hypotheses in the form of a semantic net, the system may be able to point out specific characters which conflict with known character state distributions.

Most identification devices can be easily updated if monophyletic groups of taxa are added. For example, in a dichotomous key or an expert system, couplets or rules are simply inserted to accommodate the new

group. Inserting individual taxa or taxa which are recognized by polythetic criteria can be more problematical. To make such a modification to a dichotomous key often requires structural changes. One of the advantages of computerized keys (such as ONLIN3 and XPER) is the separation of the database from algorithms that perform operations on the data. In such a system, the user can easily update the database. In an ideal expert system, the knowledge base (what the system knows) is separate from the inference engine (which determines how that knowledge is used) (Davis, 1986). In practice, rule-based expert systems mix elements of the knowledge base and reasoning strategy within rules, making it nontrivial to update such systems. Expert systems that use other forms of knowledge representation besides rules (e.g., frames and semantic nets) do a much better job separating the domain knowledge from the reasoning rules. In SYSTEX, though, there is a trade-off between complexity of program structure and the other advantages noted for expert systems. However, since the individual rules in an expert system are independent of one another, updating an expert system to accommodate new taxa requires modification of only a subset of the rule base. The knowledge base in SYSTEX, for example, contains facts, flow of control rules, elimination rules, discrimination rules, and verification rules. To add another species to SYSTEX would require modification of one factual statement to include the species in the list of recognized species names. Thereafter, some elimination rules and discrimination rules would require modification, and a few new elimination and verification rules would be required. However, the attention of the programmer would easily be focused on the relevant parts of the rule base and the overall structure of the program would not be changed.

One desirable feature of production expert systems is the ability of the inference engine to keep in memory a record of the sequence of rules processed during a session. The user of SYSTEX, for example, by responding "why" to a query for data in-

put, can directly access the logical operations of the system. The logical strategy of a well designed dichotomous key may be relatively transparent to the user, but this is by no means a general feature of this device. The logical operations of computer keys using online, matching, probabilistic, or phenetic distance methods are hidden from the user, while the user of tabular or polyclave devices essentially supplies her own sequence of logical operations.

The distinction between the explanatory capability of programs which use an algorithmic method for identification (e.g., polyclave or online keys) and methods which use a heuristic methodology (expert systems) is subtle and relates to fundamental differences in the structure of each. Algorithmic programs can provide a history of their transactions, and most polyclave and online programs will list the known character states of taxa and indicate by which states an unknown taxon differs. An expert system can explain (by backtracking through the rules which have been executed) the reasoning which has led to the present situation, and it can describe what it is currently attempting to do (by presenting the current rule) at any point in a session.

An expert system can be designed to learn from experience by internal modification of its own rule base during consultations with users. This is an exciting prospect, however, such a process would require rigid verification by the domain expert to avoid incorporation of incorrect data. Certain other computer algorithms for identification have included similar features (e.g., see Rypka, 1975).

SUMMARY AND CONCLUSIONS

Our experience with SYSTEX has convinced us that expert systems techniques offer a set of advantages for identification not found in any other single identification device (Table 1). In designing SYSTEX and constructing the rule base, we found that it is easier for at least one systematist (J.B.W.) to describe the logical thought process and criteria for species identification to a knowledge engineer for encoding into

a rule base, than it is to design a dichotomous key. Once the general structure of the expert system has been formulated, the transition from a taxa by characters matrix to a rule base is straightforward.

The basic structure of SYSTEX, or some modification of it, could serve as a template for the design of similar applications in diverse taxa. However, the systematist contemplating the design of an expert system should choose a software package with command of sufficient computer memory to handle the specific application. For example, with the M1 product we found that the working memory available to the program (256 kilobytes) was consumed by the knowledge base required for a relatively modest pilot application, in this case the identification of 12 species. Several expert system shells are now available for use on microcomputers and minicomputers. Once an expert system has been designed and tested on a particular computer, it is possible to distribute the program in executable machine-language for the use of systematists with access to computers with compatible operating systems. However, because many expert system shells (like M1) operate as interpreters, not compilers, distribution of developed systems often will require recipients to purchase the same expert system shell.

The expert system approach will be most profitably applied to groups in which the demand for identifications is substantial, expertise for identifications is scarce, and the accuracy of identifications is critical. They are well suited to taxa in which identification by traditional means or other computerized methods is difficult. Expert systems are particularly appropriate when identification criteria are polythetic, or when the probability of encountering undescribed taxa is high. We do not predict that expert systems will replace dichotomous or tabular keys, as these remain the only devices convenient for publication. For groups in which the need for identifications is occasional, the dichotomous or tabular key will continue to be widely used.

Expert systems do have some disadvantages. At the present time the expertise re-

quired to design and update expert systems (knowledge engineering) is scarce. Few systematists will wish to take the time to learn an artificial intelligence language (e.g., LISP or PROLOG) or to become fluent with an expert system design package. The continued development of expert systems for taxonomic identification will be possible only through collaboration between systematists and artificial intelligence researchers.

ACKNOWLEDGMENTS

We would like to thank Douglas Loh for assistance and advice during the early stages of this project and for his critical review of the manuscript. We thank also Horace Burke, Bob Wharton, Tim Friedlander, and two anonymous reviewers for reading the manuscript and for their valuable comments and suggestions. This paper is Technical Article No. 21316 from the Texas Agricultural Experiment Station.

REFERENCES

- BOHART, R. M. 1961. The art and practice of key-making in entomology. *Verh. XI Int. Kong. Entomol.*, Wien, 1960, 1:17-19.
- BUCHANAN, B. G., AND E. H. SHORTLIFFE. 1984. Rule based systems: The MYCIN experiments of the Stanford Heuristic Programming Project. Addison-Wesley Publishing Company, Menlo Park, California.
- DALLWITZ, M. J. 1974. A flexible computer program for generating identification keys. *Syst. Zool.*, 23: 53-57.
- DALLWITZ, M. J. 1978. User's guide to KEY—A computer program for generating identification keys. CSIRO Div. Entomol. Rep. No. 4. 16 pp.
- DAVIS, R. 1986. Knowledge-based systems. *Science*, 231:957-963.
- DAVIS, R., AND J. KING. 1984. The origin of rule-based systems in AI. Pages 20-52 in *Rule based expert systems: The MYCIN experiments of the Stanford Heuristic Programming Project*. Addison-Wesley Publishing Company, Menlo Park, California.
- DUDA, R. O., AND J. G. GASCHNIG. 1981. Knowledge-based expert systems come of age. *Byte*, 6(9):238-278.
- DUDA, R. O., J. G. GASCHNIG, AND P. E. HART. 1979. Model design in the PROSPECTOR consultant system for mineral exploration. Pages 153-167 in *Expert Systems in the micro-electronic age* (D. Michie, ed.). Edinburgh University Press, Edinburgh.
- FEIGENBAUM, E. A., B. G. BUCHANAN, AND J. LEDERBERG. 1971. On generality and problem solving: A case study using the DENDRAL program. Pages 165-190 in *Machine intelligence 6* (B. Meltzer and D. Michie, eds.). American Elsevier, New York.
- FORGET, P. M., J. LEBBE, H. PUIG, R. VIGNES, AND M. H. HIDEUX. 1986. Microcomputer-aided identification: An application to trees from French Guiana. *Bot. J. Linn. Soc.*, 93:205-223.
- GYLLENBERG, H. G. 1963. A general method for deriving determinative schemes for random collections of microbial isolates. *Ann. Acad. Sci. fenn. Ser. A, IV Biologica*, 69:5-22.
- GYLLENBERG, H. G., AND T. K. NIEMELÄ. 1975. New approaches to automatic identification of micro-organisms. Pages 121-136 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- HALL, A. V. 1975. A system for automatic key forming. Pages 55-63 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- KRAFT, A. 1984. XCON: An expert configuration system at Digital Equipment Corporation. Pages 41-49 in *The AI business* (P. H. Winston and K. Pendergast, eds.). MIT Press, Cambridge, Massachusetts and London.
- LEBBE, J. 1984. Manuel d'utilisation de logiciel XPER. Microapplication, Paris.
- METCALF, Z. P. 1954. The construction of keys. *Syst. Zool.*, 3(1):38-45.
- MORSE, L. E. 1971. Specimen identification and key construction with time-sharing computers. *Taxon*, 20:269-282.
- MORSE, L. E. 1974. Computer programs for specimen identification, key construction, and description printing using taxonomic data matrices. *Publ. Mus. Mich. State Univ. Biol. Ser.*, 5:1-128.
- MORSE, L. E. 1975. Recent advances in the theory and practice of biological specimen identification. Pages 11-52 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- NEWELL, I. M. 1970. Construction and use of tabular keys. *Pac. Insects*, 12(1):25-37.
- NEWELL, I. M. 1972. Tabular keys—Further notes on their construction and use. *Trans. Conn. Acad. Arts Sci.*, 44:259-267.
- NOYES, J. S., AND M. HAYAT. 1984. A review of the genera of Indo-Pacific Encyrtidae (Hymenoptera: Chalcidoidea). *Bull. Br. Mus. (Nat. Hist.) Entomol.*, 48(3):131-395.
- PANKHURST, R. J. 1971. Botanical keys generated by computer. *Watsonia*, 8:357-368.
- PANKHURST, R. J. 1975. Identification by matching. Pages 79-91 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- PANKHURST, R. J. 1978. *Biological identification*. Edward Arnold Ltd., London.
- PANKHURST, R. J. 1983. An improved algorithm for finding diagnostic taxonomic descriptions. *Math. Biosci.*, 65:209-218.
- PANKHURST, R. J. 1984. Online identification program, version 4. British Museum (Natural History), London.

- PANKHURST, R. J., AND R. R. AITCHISON. 1975. An online identification program. Pages 181–192 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- PAYNE, R. W. 1975. Genkey: A program for constructing diagnostic keys. Pages 65–72 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- QUINLAN, J. R. 1982. Discovering rules by induction from large collections of examples. Pages 192–207 in *Introductory readings in expert systems* (D. Michie, ed.). Gordon and Breach Science Publishers, New York, London, and Paris.
- ROSS, G. J. S. 1975. Rapid technique for automatic identification. Pages 93–102 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- RYPKA, E. W. 1975. Pattern recognition and microbial identification. Pages 153–180 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special Volume No. 7. Academic Press, London, New York.
- STONE, N. D., R. N. COULSON, R. E. FRISBIE, AND D. K. LOH. 1986. Expert systems in entomology: Three approaches to problem solving. *Bull. Entomol. Soc. Am.*, 32:161–166.
- WILLCOX, W. R., AND S. P. LAPAGE. 1975. Methods used in a program for computer-aided identification of bacteria. Pages 103–119 in *Biological identification with computers* (R. J. Pankhurst, ed.). The Systematics Association Special volume No. 7. Academic Press, London, New York.
- WILSON, J. B., AND T. R. PARTRIDGE. 1986. Interactive plant identification. *Taxon*, 35:1–12.

Received 17 March 1987; accepted 9 June 1987.